TRIANGULATION OF NURBS SURFACES

Jamshid Samareh-Abolhassani*

1 Abstract

A technique is presented for triangulation of NURBS surfaces. This technique is built upon an advancing front technique combined with grid point projection. This combined approach has been successfully implemented for structured and unstructured grids.

2 Introduction

Computer Aided Design (CAD) systems typically represent the surfaces of aerodynamic vehicles with a set of parametric surfaces such as NonUniform Rational B-Splines (NURBS). Then, CFD surface grids are generated on these NURBS surfaces. A surface grid can be generated either in a parameter space or on an approximated/simplified NURBS surface. Generating surface grid in a parameter space is very common in structured grid generation. This approach has two serious restrictions. The first restriction is that the choice of surface parameterization affects the CFD surface grid. As shown in [1], a poor parameterization may cause the CFD surface grid to be highly skewed. There are several ways to alleviate this problem which have been discussed in great detail in [1]. The second limitation is that a CFD surface grid can not be generated over several overlapping NURBS surfaces. This is the most serious restriction.

In the second method , the NURBS surfaces are approximated by a few smaller bi-linear patches. Then, the surface grid is generated on these bi-linear patches. This method is quite easy to implement, and it avoids the problems associated with surface parameterization. However, the resulting surface grid is close but not on the original NURBS surfaces. This problem can be alleviated by projecting the resulting surface

*Computer Scientist, Computer Sciences Corporation, Hampton, Virginia 23666.

grid onto the original NURBS surfaces. After the first projection, the projected grids may need to be smoothed and projected again. The projection techniques described here can be used for structured grid generation as well.

In this study, the NURBS surfaces are first approximated by a set of smaller bilinear patches. Then, an advancing front technique [2] is used to generate surface grid on these patches. And, finally this surface grid is projected back onto the NURBS surfaces. In the following sections, the results of surface triangulation in a parameter space is presented, the techniques for projecting a point on a surface is described, and finally results are summarized.

3 Triangulation in a Parameter Space

Surface triangulation in a parameter space is similar to structured-surface grid generation. There are four steps in generating a surface-grid in a parameter space. In the first step, the surface is mapped to a parameter space using one of the techniques described in [1]. The second step is to construct edges of the surface grid in the physical space and map them to the parameter space. In the third step, the interior grid is generated using the advancing front technique as described in [3]. Finally, the grid is transformed from the parameter space to the physical space.

Surfaces generated by a CAD program may have a very poor parameterization. This could be due to the required CAD operations such surface concatenation, trimming, splitting, conversion, ...etc. Surface properties such as curvature, nonuniform spacing (parametrization), aspect ratio and grid orthogonality have influence on the quality of surface grid generated in a parameter space as demonstrated in [1]. Results from structured grid generation can be extended to unstructured grid generation in regard to effect of parameterization. For smooth surfaces, the surface triangulation may be generated in any of the parameter spaces as discussed in [1]. However, if the surface is not smooth or it is skewed, then the surface triangulation should be generated in an arclength parameter space. The FELISA code [3] is used to test the surface triangulation in a uniform parameter space. The code is very easy to use and it generates a very good surface triangulation provided that the surface has a good parameterization. Three test cases are presented here, and in all three cases a uniform spacing is speci-

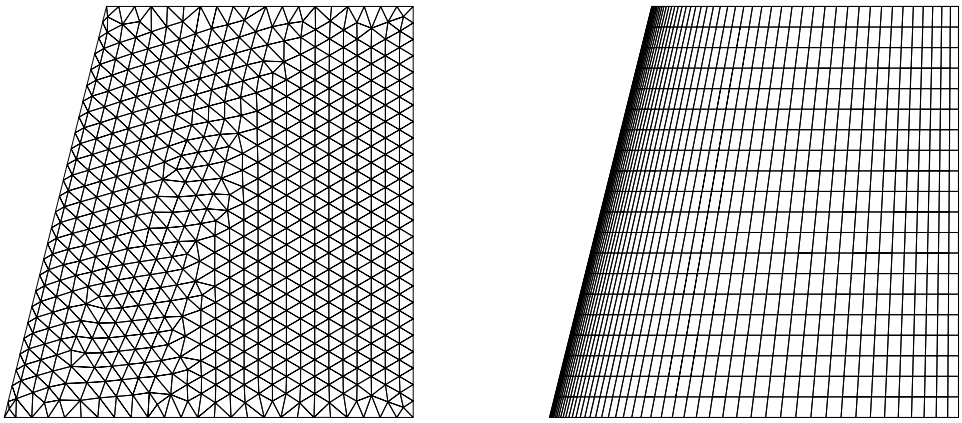

Figure 1: Triangulation of a Wing Section With Smooth Parameterization.

fied for the entire domain. Figures 1-2 show two different surface triangulations for a wing section in a uniform parameter space. Figure 1 shows surface triangulation for a surface with "good parameterization". This surface grid is very smooth. Figure 2 shows surface triangulation for a surface with "poor parameterization". This surface triangulation is very smooth except where the surface parameterization is not continuous. These two examples are for a nonplanar surface. In order to separate the effect of curvature, a planar surface is selected for the next example. Figure 3 shows the surface triangulation for a square plane in a uniform parameter space. This surface has a "poor parameterization". Again, the surface triangulation is very smooth except where the surface parameterization is not continuous.

4 Projecting a Point onto a Surface

Projecting a point on a curve has been discussed in [4]. In this section, projecting a point on a surface will be discussed. Surfaces are generally represented by parametric

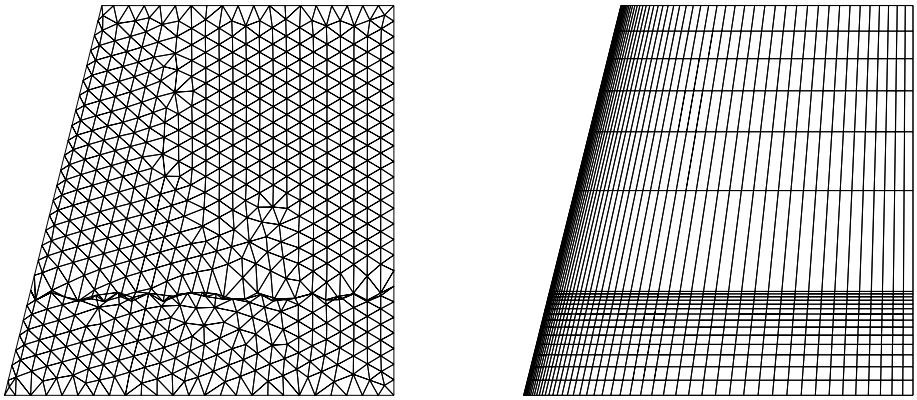

Figure 2: Triangulation of a Wing Section With Discontinuous Parameterization.

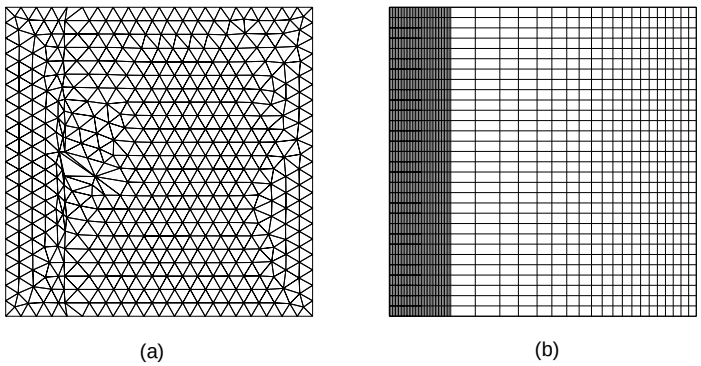

Figure 3: Triangulation of a Square Plane With Discontinuous Parameterization.

splines, e.g., NURBS, as,

$$\vec{R}(\vec{u}) = \{x(\vec{u}), y(\vec{u}), z(\vec{u})\}^T,$$

$$\vec{u} = \{u_1, u_2\}^T, \in [(a, b), (c, d)],$$

where $\vec{u}$ are the parameters which have no geometric significance. However, for a constant u_2, as u_1 increases, the point $\vec{R}(\vec{u})$ always moves from one side of the surface to the other side. The process of projecting a point, $\vec{r}$, on a surface, $\vec{R}(\vec{u})$, can be achieved by finding $\vec{u}$ such that distance, d, between $\vec{r}$ and $\vec{R}(\vec{u})$ is minimum and $\vec{u}$ is constrained to be $\in [(a, b), (c, d)]$. Distance, d, can be written in terms parameter $\vec{u}$ as,

$$d^2(\vec{u}) = f(\vec{u}) = |\vec{R}(\vec{u}) - \vec{r}|^2. \tag{1}$$

In order to find the minimum distance, Eq. 1 must be minimized with respect to $\vec{u}$. This can be accomplished by setting the gradient of f, $\nabla f(\vec{u})$, equal to zero, as

$$\nabla f(\vec{u}) = G_i(\vec{u}) = \frac{\partial f(\vec{u})}{\partial u_i} = 0, \tag{2}$$

$$G_i(\vec{u}) = \frac{\partial \vec{R}(\vec{u})}{\partial u_i} \cdot \{\vec{R}(\vec{u}) - \vec{r}\}.$$

The above nonlinear system of equations must be solved for $\vec{u}$. Three cases are discussed here: (1) projection on a three-dimensional triangle element, (2) projection on a bilinear patch, and (3) projection on a NURBS surface. As shown in Fig. 4, a three-dimensional triangle element can be represented in terms of its parametric coordinates as

$$\vec{R}(\vec{u}) = \vec{R}_1 u_1 + \vec{R}_2 u_2 + \vec{R}_3 u_3,$$

$$u_1 + u_2 + u_3 = 1,$$

or

$$\vec{R}(\vec{u}) = (\vec{R}_1 - \vec{R}_3)u_1 + (\vec{R}_2 - \vec{R}_3)u_2 + \vec{R}_3. \tag{3}$$

Combining Eqs. 2-3 results in a set of linear equations which can be rewritten as,

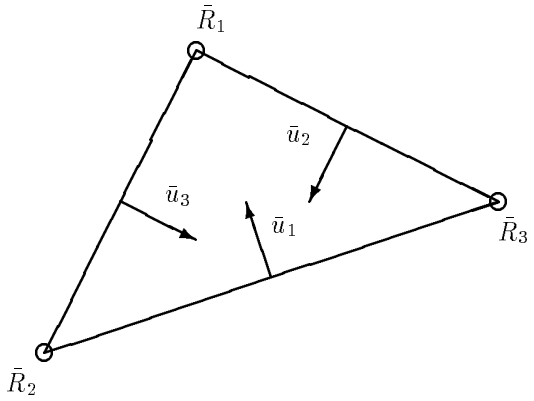

Figure 4: Parameter for a Triangular Element

$$
\begin{aligned}
(S_2 \cdot S_2)u_1 + (S_1 \cdot S_2)u_2 &= (S \cdot S_2) \\
(S_2 \cdot S_1)u_1 + (S_1 \cdot S_1)u_2 &= (S \cdot S_1) \\
u_1 + u_2 + u_3 &= 0.
\end{aligned} \tag{4}
$$

For triangular elements, the above equations are solved for $\vec{u}$ (see Fig. 5 for definitions of S_1, S_2 and S). The projected point is inside the triangle if

$$0 \leq u_i \leq 1, \quad \text{and} \quad 0 \leq u_1 + u_2 + u_3 \leq 1,$$

otherwise it is outside. If the projected point is outside of the triangular element, the parameters, u_i, must be clipped as

$$u_i = \min\{u_i, 1\}, \qquad ui = \max\{u_i, 0\}. \tag{5}$$

The second case is for projecting a point onto a bilinear surface, where the surface is approximated by a set of structured points. A bilinear surface can be decomposed into a set of bilinear patches, and each patch is approximated in terms of its parameters (see Fig. 6) as

$$
\begin{aligned}
\vec{R}(u_1, u_2) = (1-u_1)(1-u_2)\vec{R}_{i,j} \quad &+ \quad (1-u_1)u_2\vec{R}_{i,j+1} + \\
u_1(1-u_2)\vec{R}_{i+1,j} \quad &+ \quad u_1u_2\vec{R}_{i+1,j+1}.
\end{aligned} \tag{6}
$$

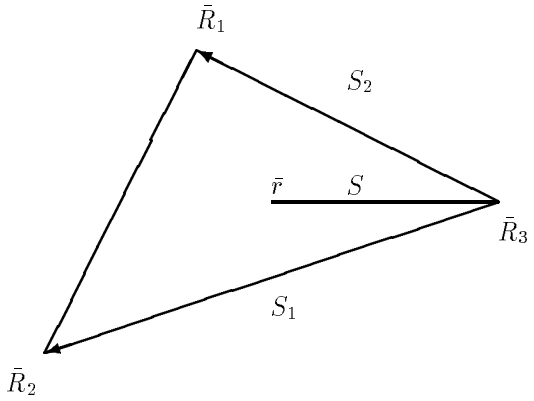

Figure 5: Projection on a Triangular Element

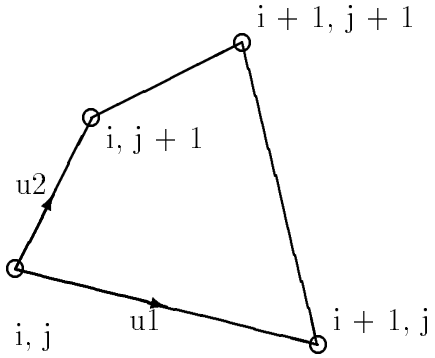

Figure 6: Parameter Space for a Bi-Linear Patch

Combining Eqs. 2 and 6 yields the following system of nonlinear equations,

$$\frac{\partial \vec{R}}{\partial u_1} \cdot \{\vec{R}(\vec{u}) - \vec{r}\} = 0, \qquad \frac{\partial \vec{R}}{\partial u_2} \cdot \{\vec{R}(\vec{u}) - \vec{r}\} = 0, \tag{7}$$

where,

$$\frac{\partial \vec{R}}{\partial u_1} = (1-u_2)(\vec{R}_{i+1,j} - \vec{R}_{i,j}) + u_2(\vec{R}_{i+1,j+1} - \vec{R}_{i,j+1}),$$

$$\frac{\partial \vec{R}}{\partial u_2} = (1-u_1)(\vec{R}_{i,j+1} - \vec{R}_{i,j}) + u_1(\vec{R}_{i+1,j+1} - \vec{R}_{i+1,j}).$$

Equation 7 is solved by the Newton-Raphson method. It takes an average of 5 iterations to converge. The method requires an initial guess which is found by sampling the surface at various locations (see Fig. 7).

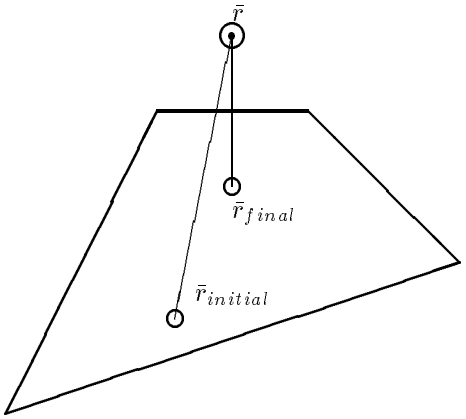

Figure 7: Projection Process for a surface

The third case is for projecting a point onto a NURBS surface. A NURBS surface can be expressed as

$$\vec{R}(\vec{u}) = \frac{\sum_{i=0}^{n}\sum_{j=0}^{m} N_{i,p}(u_1)N_{j,q}(u_2)w_{i,j}\vec{P}_{i,j}}{\sum_{i=0}^{n}\sum_{j=0}^{m} N_{i,p}(u_1)N_{j,q}(u_2)w_{i,j}}, \tag{8}$$

where $n+1$ and $m+1$ are the number of control points in i and j directions, respectively. The $\vec{P}_{i,j}$ are control points (forming a control polygon), $w_{i,j}$ are the weights, and $N_{i,p}(u_1)$ and $N_{j,q}(u_2)$ are the p-th and q-th degree B-spline basis function defined on the non-periodic and nonuniform knot vector [5], respectively. The $N_{\ell,\rho}$ is defined as

$$N_{\ell,\rho}(u) = \frac{u-u_\ell}{u_{\ell+\rho}-u_\ell}N_{\ell,\rho-1} + \frac{u_{\ell+\rho+1}-u}{u_{\ell+\rho+1}-u_{\ell+1}}N_{\ell+1,\rho-1}.$$

$$N_{\ell,0}(u) = \begin{cases} 1 & \text{if } u_\ell \le u < u_{\ell+1} \\ 0 & \text{otherwise} \end{cases}$$

Combining Eqs. 2 and 8 yields a system of nonlinear equations similar to Eq. 7 which is solved by the Newton-Raphson method. For NURBS surfaces, the Newton-Raphson method is similar to the variable metric method [6] which is based on a Taylor series expansion of f about $\vec{w}$. This method is iterative and at each iteration the solution,

$\vec{w}^n$, is replaced with a corrected solution, $\vec{w}^{n+1}$. The corrected solution is obtained by a Taylor series expansion about $\vec{w}^n$ as

$$f(\vec{w})^{n+1} = f(\vec{w})^n + G_i(\vec{w})^T \Delta\vec{w} + \frac{1}{2}\Delta\vec{w}^T[H]\Delta\vec{w} + \cdots \tag{9}$$

In the optimization terminology, the $G_i(\vec{w})$ is referred to as the search direction (gradient vector) and the $[H]^{-1}$ as the step size (Hessian matrix). They are defined as

$$G_i(\vec{w}) = \nabla f(\vec{w}) = \frac{\partial f(\vec{w})}{\partial u_i} = \frac{\partial \vec{R}(\vec{w})}{\partial u_i} \cdot \{\vec{R}(\vec{w}) - \vec{r}\}, \tag{10}$$

$$H_{i,j}(\vec{w}) = \nabla(\nabla f(\vec{w})) = \frac{\partial^2 f(\vec{w})}{\partial u_i \partial u_j}, \quad 1 \leq i, j \leq 2. \tag{11}$$

In order to minimize $f(\vec{w})^n$, Eq. 9 is differentiated with respect to $\Delta\vec{w}$ and set equal to zero as,

$$[H][\Delta\vec{w}] = -[G], \qquad \vec{w}^{n+1} = \vec{w}^n - \Delta\vec{w}. \tag{12}$$

The function $f(\vec{w})$ is minimum when $\frac{\partial f(\vec{w})}{\partial \vec{w}}$ is zero and the $H_{i,j}(\vec{w})$ is positive definite. The step-size has to be monitored during convergence. It is possible for this method to oscillate or even to diverge [7]. This occurs if within two consecutive iterations, the search directions are orthogonal (at each iteration the direction of search reverses),

$$\nabla f(\vec{w}^{n+1})^T \nabla f(\vec{w}^n) = 0.$$

The Newton-Raphson method converges quadratically provided that the matrix H is not singular and a sufficiently accurate initial guess is known. In order to find an accurate initial guess, the surface is subdivided into a set of bilinear patches. First, a patch with its center closest to the "projected point" is found. Then, the patch is subdivided into two triangles. The point is projected to both triangles, and the closet triangle is found. The parameters of the projected point are found using Eq. 4. These parametric coordinates may need to be rotated to match the patch coordinates. Then, the corrected guess is used as an initial guess for the Newton-Raphson method, Eq. 12.

5 Results and Discussions

Two surface triangulations are projected in this study. The first surface triangulation is projected onto a bilinear surface. Figure 8 shows the result of projecting an unstructured

grid on a bilinear surface. Figure. 9 shows the surface contours before (solid-line) and after (dash-line) projection. Figure 10 shows the projected surface triangulation for an X-15 configuration. This grid has been projected onto ten NURBS surfaces. Figure 11 shows the surface contours before (solid-line) and after (dash-line) projection. In both examples the projected surface grids were not distorted.

By using the projection technique described above, it is possible to project structured and unstructured grids onto several overlapping CAD surfaces.

References

[1] Samareh-Abolhassani, Jamshid, and Stewart, John E., "*Surface Grid Generation in a Parameter Space,* " AIAA-92-2717, 1992.

[2] Parikh, P. Pirzadeh, S., and Löhner, T., "*A Package for 3-D Unstructured Grid Generation Finite-Element Flow Solution and Flow Field Visualization,*" NASA CR-182090, September 1990.

[3] Peraire, J., Peiro, J., Formaggia, L., Morgan, K., Zienkiewics, O., "*Finite Element Euler Computations in Three Dimensions,*" International Journal for Numerical Methods in Engineering, vol 26, pp. 2135-2159, 1988.

[4] Samareh-Abolhassani, Jamshid, "*Unstructured Grid on NURBS Surfaces,*" AIAA-93-3454, 1993.

[5] Tiller, Wayne, "*Theory and Implementation of Non-Uniform Rational B-Spline Curves and Surfaces: class notes for NASA Langley,*" August 1991.

[6] Press, W. H., Flannery, B. P., Teukolsky, S. A., Vetterling, W. T.,"*Numerical Recipes: The Art of Scientific Computing,*", Cambridge University Press, 1989.

[7] Conte, S. D., de Boor, C.,"*Elementary Numerical Analysis: An Algorithmic Approach,*" Third Edition, McGraw-Hill, New York, 1980.

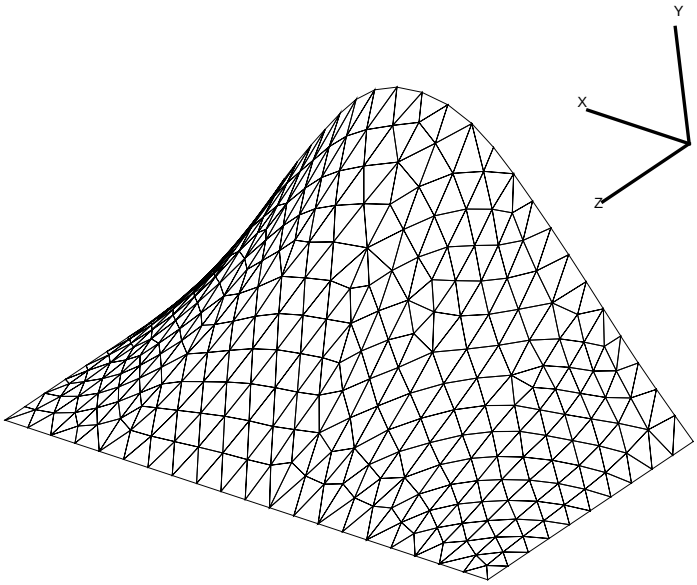

Figure 8: Projection for a Bilinear Surface

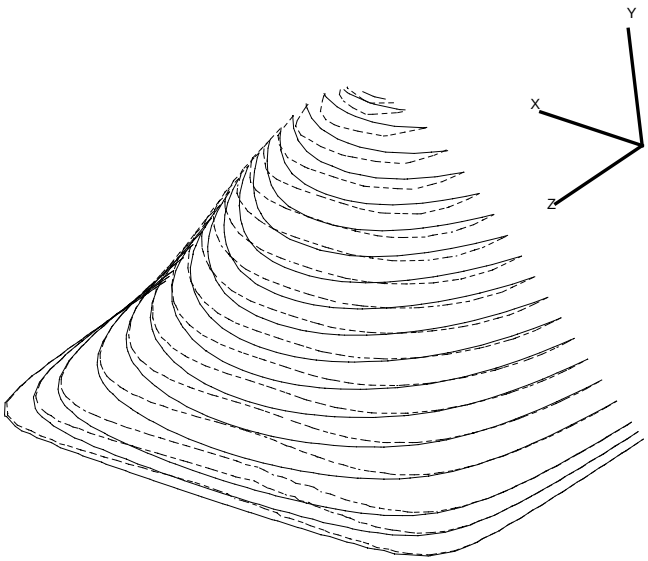

Figure 9: Projection for a Bilinear Patch

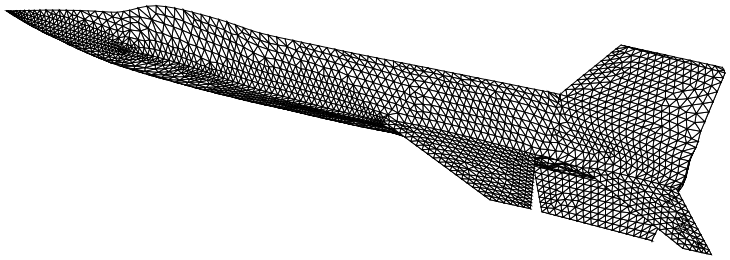

Figure 10: Triangulation for an X-15

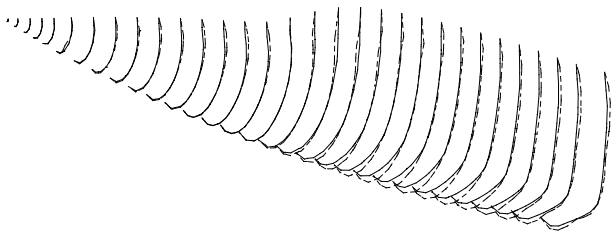

Figure 11: Surface Contours of an X-15